Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Claims 1-40 (cancelled).

Claim 41 (currently amended):   A method for providing mutual exclusion for a resource in a computer system having a plurality of processes, the method comprising:

maintaining a resource lock for each process requiring access to the resource, the resource lock having a plurality of fields requiring initialization in order for the process to access the resource, the plurality of fields including an owner indicator field for indicating an owner process for the resource;

receiving, by a first process, an inquiry from a second process inquiring whether the first process owns the resource;

determining, by the first process, an owner process for the resource other than the first process; [[and]]

creating a ghost lock for the first process, wherein the ghost lock is a partial instantiation of a resource lock having at least the owner indicator field initialized to indicate the owner process determined for the resource but having less than all fields initialized, and wherein the ghost lock allows the first process to identify the owner process for the resource without first sending an inquiry message to determine the owner process;

sending, by the second process, an inquiry to a third process inquiring whether the third process owns the resource;

receiving, by the second process, a response from the third process indicating whether the third process is the owner process for the resource;

determining, by the second process, that the second process is the owner process for the resource, if the response indicates that the third process is not the owner process for the resource;

creating an owner lock for the second process if the second process is the owner process for the resource, wherein the owner lock is a resource lock having all fields initialized and the owner indicator field indicating that the second process is the owner process for the resource; and

creating a reference lock for the second process if the third process is the owner process for the resource, wherein the reference lock is a resource lock having all fields initialized and the owner indicator field indicating that the third process is the owner process for the resource.

Claims 42-45 (canceled).

Claim 46 (currently amended):   The method of claim [[45]] 41, further comprising:

sending, by the second process, an owner notification message to the first process indicating the owner process for the resource, the owner process being one of the second process and the third process.

Claim 47 (previously presented): The method of claim 46, wherein determining the owner process by the first process comprises:

determining the owner process for the resource based upon the owner notification message.

Claim 48 (previously presented): The method of claim 41, further comprising:

determining that the first process requires access to the resource;

identifying, by the first process, the owner process for the resource using the ghost lock; and

sending, by the first process, a request message to the owner process requesting access to the resource without first sending an inquiry message to determine the owner process.

Claim 49 (previously presented): The method of claim 48, wherein identifying the owner process for the resource using the ghost lock comprises:

finding the ghost lock among a plurality of resource locks based upon a resource identifier; and

obtaining the owner process from the owner indicator field of the ghost lock.

Claim 50 (previously presented): The method of claim 48, further comprising:

converting the ghost lock to a reference lock by initializing all uninitialized fields of the lock.

Claims 51-64 (canceled).

Claim 65 (currently amended): A computer system comprising a plurality of processes sharing a resource, wherein:

a resource lock is maintained for each process requiring access to the resource, the resource lock having a plurality of fields requiring initialization in order for the process to access the resource, the plurality of fields including an owner indicator field for indicating an owner process for the resource;

a first process receives an inquiry from a second process inquiring whether the first process owns the resource;

the first process determines an owner process for the resource other than the first process; [[and]]

a ghost lock is created for the first process, wherein the ghost lock is a partial instantiation of a resource lock having at least the owner indicator field initialized to indicate the owner process determined for the resource but having

less than all fields initialized, and wherein the ghost lock allows the first process to identify the owner process for the resource without first sending an inquiry message to determine the owner process;

the second process sends an inquiry to a third process inquiring whether the third process owns the resource;

the second process receives a response from the third process indicating whether the third process is the owner process for the resource;

the second process determines that it is the owner process for the resource, if the response indicates that the third process is not the owner process for the resource;

an owner lock is created for the second process if the second process is the owner process for the resource, wherein the owner lock is a resource lock having all fields initialized and the owner indicator field indicating that the second process is the owner process for the resource; and

a reference lock is created for the second process if the third process is the owner process for the resource, wherein the reference lock is a resource lock having all fields initialized and the owner indicator field indicating that the third process is the owner process for the resource.

Claims 66-69 (canceled).

Claim 70 (currently amended):   The computer system of claim [[69]] 65, wherein:

the second process sends an owner notification message to the first process indicating the owner process for the resource, the owner process being one of the second process and the third process.

Claim 71 (previously presented): The computer system of claim 70, wherein the first process determines the owner process for the resource based upon the owner notification message.

Claim 72 (previously presented): The computer system of claim 65, wherein:

the first process identifies the owner process for the resource using the ghost lock upon requiring access to the resource and sends a request message to the owner process requesting access to the resource without first sending an inquiry message to determine the owner process.

Claim 73 (previously presented): The computer system of claim 72, wherein the first process identifies the owner process using the ghost lock by finding the ghost lock among a plurality of resource locks based upon a resource identifier and obtaining the owner process from the owner indicator field of the ghost lock.

Claim 74 (previously presented): The computer system of claim 72, wherein the first process converts the ghost lock to a reference lock by initializing all uninitialized fields of the lock.